Queue Editor

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	1024 megabytes

Little Cyan Fish has two queues Q_1 and Q_2, with sizes m and $m+1$, respectively. Initially, both of the queues are empty.

He needs to sequentially add several elements to both queues simultaneously. Formally, he needs to find a sequence $c_1, c_2, \ldots, c_k$, and add $c_1, c_2, \ldots, c_k$ to both queues in order.

When an element is added, if it is already in the queue, nothing happens; otherwise, the element is added to the end of the queue. If the number of elements in the queue exceeds its size, the element at the front of the queue is popped out.

Now, Little Cyan Fish gives you the final state of the two queues. The elements in these two queues, from front to back, are $a_1, a_2, \ldots, a_m$ and $b_1, b_2, \ldots, b_{m+1}$, respectively.

Little Cyan Fish wants to know if it is possible to construct a sequence c that satisfies this condition, with the length of the sequence not exceeding $50 \cdot m$. He knows that under the constraints of the problem, if a solution exists, its length will not exceed $50 \cdot m$.

Input

There are multiple test cases in a single test file. The first line of the input contains an integer T ($T \geq 1$) indicating the number of test cases. For each test case:

The first line of the input contains an integer m ($m \geq 4$), indicating the sizes of the two queues.

The next line of the input contains m integers $a_1, a_2, \ldots, a_m$ ($1 \leq a_i \leq 10^9$). It is guaranteed that the elements in a are all distinct.

The following line contains $m+1$ integers $b_1, b_2, \ldots, b_{m+1}$ ($1 \leq b_i \leq 10^9$). It is guaranteed that the elements in b are all distinct.

It is guaranteed that the sum of m over all test cases will not exceed 2×10^4.

Output

For each test case, if it is impossible to choose the subsequence, output a single line "`No`".

Otherwise, the first line of the output should contain the word "`Yes`". Then, the next line of the output should contain a single integer k ($1 \leq k \leq 50 \cdot m$), indicating the length of the sequence.

Then output a line with k integers $c_1, c_2, \ldots, c_k$ ($1 \leq c_i \leq 10^9$), indicating the constructed sequence.

If there are multiple possible solutions, you may output any of them.

Example

standard input	standard output
3 4 2 3 4 5 1 2 3 4 5 5 1 3 5 7 9 2 4 6 8 10 11 4 100 200 300 400 100 200 300 400 500	Yes 5 1 2 3 4 5 No Yes 9 100 200 300 400 500 100 200 300 400